GALGOTIAS UNIVERSITY
(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

Hindi handwriting recognition

Project

Submitted by

Bharat Yadav

(16SCSE101352/1613101224)

Bachelor of technology

In

Computer Science of Engineering

Under the supervision of

Pannel Member- **Dr T. Poongodi(CSE)**

Project Guide- **Dr Ganesh Gopal**

Table of contents

1.Abstract

We present a handwritten Hindi character recognition system based on different Deep learning technique. Handwritten character recognition plays an important role and possible applications in assisting technology for blind and visually impaired users, human–robot interaction, automatic data entry for business documents, etc. In this work, we propose a technique to recognize handwritten Hindi characters using deep learning approaches like Convolutional Neural Network (CNN).This is a Character Recognition System which I developed for Devanagari Script

2.Introduction

Automatic character recognition is a process that converts scanned document images into electronically understandable format. Thus, enabling computers to recognize text present in images. The latest advancements in technology have highlighted the need for robust methods of automatic character recognition.There are techniques which have been implemented for hindi character recognition as discussed in next section but there was a need of more complete and modern architecture for recognition. Thus,this article uses deep learning concepts for character recognition.

Many artificial intelligence tasks can be solved by identifying the right set of features, and then providing these features to classifier. For example, estimating the size of speaker's vocal tract is a useful feature for speaker identification, estimating pressure points and pen up and down movements are useful feature for online handwriting recognition. However, for many tasks, it is difficult to identify the right set of features. The solution to this problem is deep learning, also called end-to end learning. It is called end-toend learning because feature extraction and classification phase is automatically done,unlike traditional machine learning, where

features are to be explicitly specified. Deep architectures have provided to solutions to some well known problems of pattern recognition which are mental load classification, speech recognition, document recognition, object detection, scene classification, pedestrian detection etc.

Devanagari is an Indic script and forms a basis for over 100 languages spoken in India and Nepal including Hindi, Marathi, Sanskrit, and Maithili. It comprises of 47

.

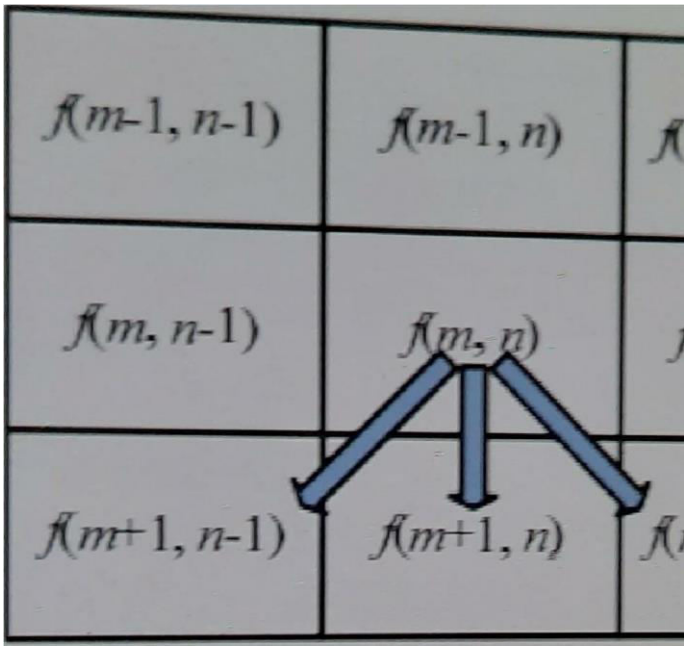
3.Proposed System

Same data sets are used for implementation by using existing work techniques i.e. SVM along with CNN. The accuracy with this method is comparatively low as compared to hybrid technique of neural network along with CNN. The proposed character segmentation algorithm consists of preprocessing, segmentation, postprocessing and post verification using SVM to validate the result has correct or not.

These steps are describing as follows..

Pre processing:- in pre processing,greyscale word image size X×Y having pixel intensity f(m,n) of the pixel location where (m,n) is binaries. Binalization is performed to reduce computation complexity of the algorithms as only two color are available for processing. Morphologically erosion operation is then performed on the binary word to join disconnect due to noise.Later word image is correct and thinning operation is applicable .

Character Segmentation:- initially,vertical projection profile of thined words images is calculative .Let,this thined image is denotted by X× Y have pixel intensity. Segmentation Path (SP), only those projection lines are retained ,which has contained value in range '0' to '2'.). When these projection lines are mapped on word image, over-segmentation occurs. This over- segmentation is emerged due to the connections or ligatures exist between characters and within the characters like ‗r', ‗m', ‗n', ‗u'. To reduce this over segmentation, some of the projection lines are reduced by applying the following conditions. (i) If difference of two projection lines is less than or equal to threshold of 3 pixels, retain the right one and remove the left one. (ii) If difference of two projection profile lines is greater than this threshold.



Searching mechanism for finding next Segmentation pixel

(i) If present pixel‘s intensity f (m, n) = 255, then pixel at location (m, n) becomes a segmentation pixel and searching mechanism will start exactly from next row and same column pixel, i.e. from the weights from graph theory are considered for assigning priorities of next rowof pixel

pixel at location $(m + 1, n)$.

(ii) If present pixel‘s intensity f (m, n) <= 255, then pixel at location (m, n) becomes a segmentation pixel and searching mechanism will start from next row and any of the next column pixels, i.e. from the pixels at locations $(m + 1, n - 1)$, $(m + 1, n)$ or $(m + 1, n + 1)$. Here, these pixel positions are considered as nodes of a graph.Therefore,Euclidean distance

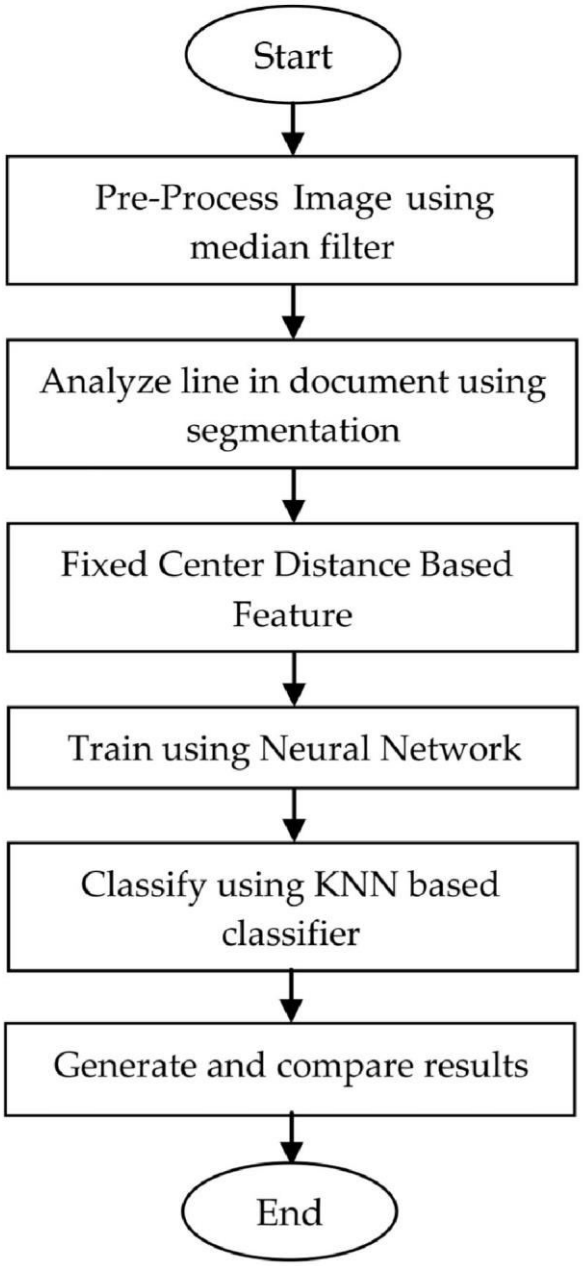
. 4) Post-Verification: Promising results are obtained for character segmentation ahead of this step. However, over-segmentation problem is still persists in open characters like _n‘, _v‘, _m‘, _w‘, _u‘. Therefore, to avoid this problem and validate the segmentation results, post-verification is performed using k-NN classifier. Suppose a training set of label pairs (xi, yi), i = 1, 2, ... m is given, where xi ϵ Rn andyiϵ {1, −1}m.Post Verification in Proposed algorithmLet (Xi, Ci) where i=1, 2……., n be data points.Xi denotes feature values &Ci denotes labels for Xi for each i.Assuming the number of classes as _c‘Ci ∈ {1, 2, 3, , c} for all values of i.Let x be a point for which label is not known, and we would like to find the label class using k-nearest neighbor algorithms.

Proposed Algorithm:

1) Calculate—$d(x, xi)$‖ i = 1, 2, … . . , n; where d denotes the Euclidean distance between the points.
2) Arrange the calculated n Euclidean distances in non-decreasing order.
3) Let k be a +ve integer, take the first k distances from this sorted list.
4) Find those k-points corresponding to these k- distances.
5)

Let ki denotes the number of points belonging to te

The overall structural design of the DFFNN Model of our proposed system with different layer.



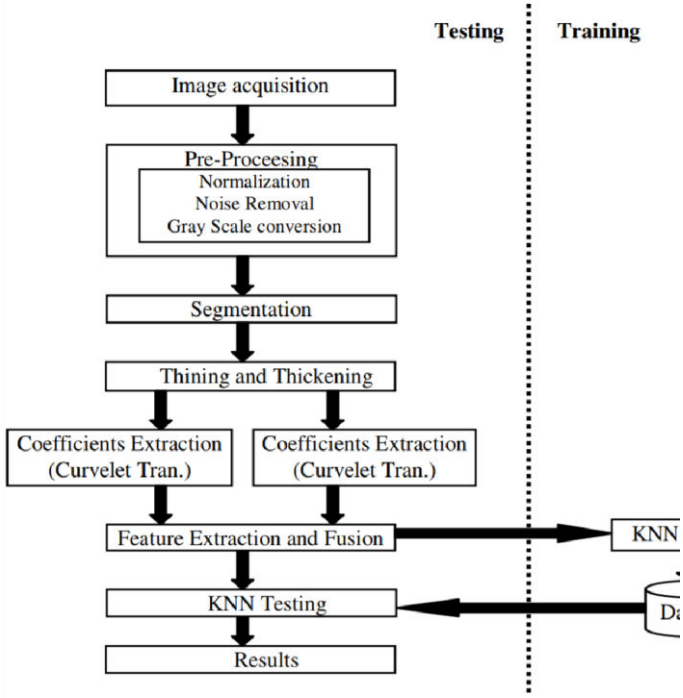


Fig. 1. Proposed system block diagram

- Install opencv , in conda -> $ conda install opencv
- HandWritingRecognition.py require data set data.csv for training and
- Application.py require devanagari_model.h5 model for classify the characters.

4. Architecture:-

CONV2D --> MAXPOOL --> CONV2D --> MAXPOOL -->FC -->Softmax--> Classification

Python Implementation:-

- Dataset- DHCD (Devnagari Character Dataset)
- Images of size 32 X 32
- Convolutional Network Support added.

Experimental Outcomes : -

➢ Experimental Values :

- Training Size : 70000 with 8 Epochs
- Accuracy : 95% ● Test Set Size : 2001
- Accuracy :

Code Requirements:-

- Test You can install Conda for python which resolves all the dependencies for machine
- Install tensorflow , in conda -> $ conda install tensorflow
- Install keras , in conda -> $ conda install keras

We used two methods to perform the experiments.

The pre-processing, learning. features used and the segmentation used is thesame for both the methods. The training and the methodto evaluate the distances between two sets of features is dif-

ferent. The two methods we used are as follows:

Method 1: In method 1, after the features were ex-tracted, they were trained using k-means. After perform-ing some simple tests, we established that k=4 per clusterwas the best parameter for k-means. While testing, the distance from the closest of the 4 centroids for a character wasconsidered to be the distance from that character.

Method 2: While using kmeans, synthesized vectors arecreated, which are the centroids of the clusters. This isgenerally not a good idea. To overcome this, we used anotherway where all the vectors of a character (22 on an average)were kept, and while testing, the distance from the closest ofall was considered to be the distance from the character. Thedifference here is that instead of comparing with synthesizedvectors, we calculate the distances with the original vectorsitself. On performing testing with both the methods fordifferent lexicon sizes, the results were as follows:The vocabulary file used in above experiments containswords of all sizes (containing 2 characters, 3 characters, etc).We also performed tests on files containing words of specificsizes. The results on these files were as follows:On comparing the results for different sized words on alexicon size of 10 words, we see that the accuracy increasesdramatically from 2 to 3 sized words, and then falls for 4and 5 sized words. This happens due to the following reasons. Words containing lesser number of characters (1 or 2)get misclassified because the character recognition moduledoesn't work as well yet. Since the words are so small, theword recognition is unable to recover. With words contaning 3 characters, even if 1 or 2 characters get misclassified,the combined effect of 3 characters helps correctly recognizethe word (the power of dictionary). With this, we expect th

6.CONCLUSION AND FUTURE SCOPE

In the available literature,many works is

done in recognization of handwritten English letters and various modification are done by this .But there is much less workto recognition of devanagri script ,the reason work to recognization of handwritten devanagri script with more and more accuracy, precision and recall capacity .the various methodologies utilized in the proposed work . However , different techniques for treating the issues of manual written English letters have been created,still a lot of research is required arrangements can be made accessible .In this research paper an improvement approach for OCR for devanagri is generated which in turns gave an improved accuracy of 97.4% in existing.

Future scope **-** this present diminished or Skelton discovering calculation relies upon the size of the picture which is certainly not a general excellent methodology towards building up a software like OCR ,so an attempted can be made by overcome this circumstances by improving the present finishing calculation which need additional time and efforts

7.<u>References</u>

1- V. N. Manjunath Aradhya, G. Hemantha Kumar,

"Principal Component Analysis and Generalized

Regression Neural Networks for Efficient Character

Recognition", © 2008 IEEE DOI 10.1109/ICETET.2008.214 .

2-Sonika Dogra and ChandraPrakash,"PEHCHAAN:HINDIHANDWRITTEN CHARACTER RECOGNITION SYSTEM BASED ON SVM", Vol. 4 No. 05 May 2012.

3- Sonal P Ajmire K. G. Bagade P.LRamteke," An Analytical Study of Handwritten Devanagari Character Recognition", Volume 5, Issue 10, October-2015

4-Abirami, S., Manjula, D.: A Survey of Script Identification techniques for Multi-Script

Document Images. International Journal of Recent Trends in Engineering 1(2), 246–249

(2009)